Amendments to the Specification:

Please amend the paragraph on page1, lines 4-11, as follows:

The subject matter of this application is related to co-pending patent applications Serial No. 09/241,991 filed February 2, 1999, by Pradeep Varma and Suresh Kumar for "Distributed Server for Real-Time Collaboration" and Serial No. 09/476,445 ~~09/___,___~~filed December 30, 1999, by Pradeep Varma for "Dynamic Clients, Dynamic Partitions, Locking and Migration Capability for Distributed Server for Real-Time Collaboration", which are assigned to a common assignee herewith. The disclosure of applications Serial No. 09/241,991 and 09/476,445 ~~09/___,___~~are incorporated herein by reference.

Please amend the paragraph which begins on page 8, line 23 as follows:

Figure 3 illustrates our peer-to-peer serialization protocol. Desk top computers 301, 302, and 303 ~~31, 32 and 33~~ and a network computer 304 ~~34~~ are connected in a peer-to-peer network; that is, there is no server, either centralized as in Figure 1 or distributed as in Figure 2. The clients, i.e., computers 301 to 304 ~~31 to 34~~, are connected to each other by first-in, first-out (FIFO) communication channels ~~35~~. The peer-to-peer serialization protocol is a timestamp and priority-based protocol. The timestamp is based on one global clock that is distributed and kept synchronized among the clients participating in a collaboration session. The clock distribution and maintenance of clock synchrony can be done by standard methods, as will be described in more detail below.

Please amend the paragraph on page 12, lines 14 to 19, as follows:

Figure 4 illustrates the serialization protocol without rollback. A test is made in decision block 401 to determine if all queues are non-empty. If not, the process waits for the queues to become nonempty in function block 402 before

returning to decision block 401. When all input queues are empty, the modification with the minimum timestamp amongst the modifications at the queue heads is chosen in function block 403.

Please amend the paragraph which begins on page 16, at line 13, as follows:

The Network Time Protocol (NTP), described by David Mills, "Internet RFC 1305: Network Time Protocol (version 3), ~~available at http://sunsite.cnlab-switch.ch/ftp/doc/standard/rfc/13xx/1305,~~ and David Mills, "Internet Time Synchronization: The Network Time Protocol", *IEEE Transactions on Communications*, vol. 39, no. 10, 1991, pp. 1482-1493, which has been developed for use on the Internet, is a popular example of a clock synchronization scheme. It has already been deployed on over 100,000 Internet hosts and routers, and provides accuracies of the order of tens of milliseconds on wide-area networks, and sub-milliseconds on local networks. NTP uses a set of primary time servers which are synchronized to highly accurate physical clocks (such as atomic clocks or satellite-based Global Positioning System (GPS)). Secondary servers and clients synchronize to these primary servers using a hierarchical subnet. The protocol peers periodically exchange messages containing their timestamps. These are used for determining the communication delays in the network, as well as clock offsets between the peer clocks. A filtering algorithm selects the best out of a moving window of clock offset samples. Then, an intersection algorithm is applied to find the best subset of servers believed to be accurate and fault-free. A combining algorithm computes the weighted average of offsets. Finally, a feedback loop controls the variable frequency oscillator of the local clock, to improve its accuracy and stability. The protocol is described in detail in Internet RFC 1305. A simplified form of this protocol, called the Simple Network Time Protocol (SNTP) is described in David Mills, Internet RFC 2030: Simple Network Time Protocol (version 4), ~~available at http://sunsite.cnlab-switch.ch/ftp/doc/standard/rfc/20xx/2030~~ . If the accuracy and

performance of the full-fledged NTP is not required, the SNTP can be used instead.

Please amend the paragraph on page 29, lines 8-16, as follows:

If the performance of the peer-to-peer serialization protocol presented in this disclosure degrades due to poor client and/or network response, then the protocol allows for a seamless, dynamic switch over to another scalable serialization. protocol that is presented in co-pending patent applications Serial Nos. 09/241,991 and 09/476,445 ~~09/___,___~~. The decision to switch to the distributed server disclosed in the co-pending patent applications Serial Nos. 09/241,991 and 09/476,445 ~~09/___,___~~can be based on an increase in the number of rollbacks faced in the peer-to-peer protocol, latency degradation of the peer-to-peer protocol, and also manual cues from the users of the system.